

FORCE EVALUATION IN PARTICLE METHODS FOR SELF-GRAVITATING MULTI-PHASE SYSTEMS

R. Capuzzo–Dolcetta

Ist. Astronomico, Univ. “La Sapienza”, via G.M. Lancisi, 29, I-00161, Rome, Italy

R. Di Lisio

Dip. Matematico, Univ. “La Sapienza”, P.le Aldo Moro 5, I-00185, Rome, Italy

P. Miocchi

Dip. Fisico, Univ. “La Sapienza”, P.le Aldo Moro 5, I-00185, Rome, Italy

A modern approach to the evolution of a mixed (stars and gas) self-gravitating system is the fully Lagrangian *particle* approach. The gaseous (particle) phase differs from the compact because the mutual force is given by the sum of gravity and pressure gradient. In this note we report of some characteristics, advantages and limitations of this approach for what regards the evaluation of forces in the system. In particular, a comparison between classic tree-code and fast multipole algorithm to evaluate gravitational forces is discussed.

1 Introduction

To represent a fluid with a particle method one can consider it as an ensemble of (smooth) particles, anyone being representative of a piece of fluid. In this scheme, each particle is mathematically characterized by a regular “kernel” which carries information on the average values of dynamical and thermodynamical quantities, as well as on their gradients. Each particle moves in the force field generated by the whole particle system, while the associated quantities evolve under their suitably regularized laws. We refer the reader to⁸ for a more detailed description of this method. In this scheme the difference between the collisional phase (gas) and the non-collisional one (stars) is that the former pseudo-particles interact via gravity and pressure, the latter via gravity only. Of course, one of the most important requirement for the method to work well is providing a good evaluation of the force field.

1.1 The “particle” fluid equations

In the smooth particle hydrodynamics (SPH) scheme the Lagrangian system of PDE governing the evolution of a self-gravitating fluid reduces to the set of

ODE:

$$\rho_i = \sum_{j=1}^N m_j \phi_{h_j}(\mathbf{r}_i - \mathbf{r}_j), \quad (1)$$

$$\dot{\mathbf{r}}_i = \mathbf{v}_i, \quad (2)$$

$$\dot{\mathbf{v}}_i = - \sum_{j=1}^N m_j (P_i/\rho_i^2 + P_j/\rho_j^2) \nabla \phi_{h_j}(\mathbf{r}_i - \mathbf{r}_j) + \nabla U_i, \quad (3)$$

$$\dot{u}_i = \sum_{j=1}^N m_j P_i / \rho_i^2 (\mathbf{v}_i - \mathbf{v}_j) \cdot \nabla \phi_{h_j}(\mathbf{r}_i - \mathbf{r}_j), \quad (4)$$

$$\nabla U_i = G \sum_{j=1}^N \frac{m_j (\mathbf{r}_i - \mathbf{r}_j)}{|\mathbf{r}_i - \mathbf{r}_j|^3} \int_0^{|\mathbf{r}_i - \mathbf{r}_j|} \phi_{h_j}(s) 4\pi s^2 ds, \quad (5)$$

$$f(P_i, \rho_i, u_i) = 0. \quad (6)$$

where the subscript labels the particles, m_i is the i -th particle mass and \mathbf{r}_i its position, with $i = 1, \dots, N$. ϕ_{h_i} denotes the symmetric kernel associated to the i -th particle whose size is h_i . The other symbols have the usual meaning. In this scheme the compact non-collisional phase is represented by particles with very small size, governed by pressureless Eq.s 3 and 4.

Let us fix our attention on the rhs of the motion Eq. 3 given by the sum of:

- (i) ∇U : body (gravity) force, i.e. a long-range (large scale) force;
- (ii) $\nabla P/\rho$: surface (pressure) force, i.e. a short-range (small scale) force.

The discretization of the density field with a set of particles introduces unavoidable local numerical fluctuations. These fluctuations acts on the evaluation of terms (i) and (ii) in two very different ways. The ∇P term (case (ii)) is, mathematically, a *differentiation* of a field and so it strongly depends on the short range variations of $P(\mathbf{r})$ (and on the density via the equation of state). On the contrary in the term (i) fluctuations tend to compensate themselves because gravitation acts on a long range. From a mathematical point of view this can be understood recalling that the gravitational field is given by a weighed *integration* on the density distribution and we know the integration *smooths out* the short range variation of the integrand.

So, the term (i) is very well approximated by particle methods, but this approximation requires a amount of computations $\propto N^2$ (as we will see), while term (ii) is often hard to handle because of fluctuations but has the advantage to require low computation weight.

2 Pressure force evaluation

As we said, the pressure term is the most delicate to handle. We discussed in² the problem to give a good particle approximation of this term. In that paper we studied the error of the SPH evaluation of the pressure-gradient field for the set of polytropic compressible fluids in spherical symmetry. In particular we have studied the error splitting it into a “modulus” and a “directional” part.

We have found that the direction of the pressure-gradient field is well fitted with large kernels while the best approximation for its modulus is attained with kernels such that every particle interacts effectively with an almost constant number of neighbours, independently of the particle position. This is very interesting because allows us to give a general rule for the choice of the kernel size (in our simulations this number is found to be $0.1N$). A good (giving an error less than 10%) direction of the pressure field is obtained simply doubling that kernel size. Unfortunately, this double-kernel scheme requires a CPU time which is almost 8 times that required by the single-kernel evaluation. A good balance between the quality of the approximation and the CPU weight is reached fixing the ‘direction’ kernel to be $\sqrt[3]{2}$ times the ‘modulus’ one.

In conclusion, a good prescription to estimate the pressure field is to evaluate the absolute value of the field with a kernel size giving a constant number of neighbouring particles, while the direction is estimated using another kernel size as described before.

The CPU time required by this procedure is less than two times that required by the usual (single-kernel) particle evaluation of the pressure field. Thus, this recipe is convenient when a parallelized code is used.

3 Gravitational force evaluation

As stated in the Introduction, the problem is to limit the CPU-time without an appreciable loss of precision.

The so called *particle-particle* calculation consists, simply, in the direct summation of the force contributed on every particle by all the other particles of the system. In this way the accuracy is only limited by the internal round-off error of the computer but the speed is minimum, being necessary to sum over $N(N-1)/2 \simeq N^2$ pairs. So the aim of both the methods we tested in³ and⁴, namely the *tree-code* and the *Fast Multipole Algorithm* (FMA), is to reduce the number of terms involved in the summation.

3.1 The comparison between tree-code and FMA

Detailed descriptions of *tree-code* and FMA can be found in ¹, ⁶ and ⁵ respectively. Their major feature is to reduce the amount of computation to $\propto N \log N$ via a multipole expansion of the potential. Moreover, FMA uses also Taylor expansions of the potential.

Performance comparison of the two methods in astrophysically realistic situations are needed to choose which of them is worth to be parallelized, with the aim to radically improve the time-space resolution of the simulations (see for this ³ and ⁴). To do this comparison we built our own optimized serial versions of the codes, that are slightly different from the “original” ones to make them more efficient in non-uniform situations where adaptivity is important. For our *tree-code* the detailed description can be found in ⁷, while for the FMA see ³.

The tests of comparison ran on an IBM Risc 6000 workstation using two different distributions of N particles. In the first corresponds to a uniform, random distribution of particles of equal masses. In the second case, we have distributed the set of particles in such a way to obtain a discretized profile of the “clumped” matter distribution known as Schuster’s ⁹ profile.

We have considered up to the *quadrupole term* in the *tree-code*, and up to the *second order term* in the multipole expansion in the FMA (the level of accuracy is in both cases 1 %). The tests did not consider the time evolution of the system: we are only interested to estimate the CPU-time spent in the evaluation of forces.

Figure 1 gives the CPU-time needed to calculate the accelerations vs. the number N of particles (the CPU-time of the *particle-particle* method is given for comparison). Both the algorithms are slower to compute forces in the non-uniform case than in the uniform one. This is obviously due to the more complicated and non-uniform spatial subdivision in clusters of particles. In both cases, and for N varying in the range we tested, the *tree-code* is faster than the FMA. So, at least for N in the range of our tests, the FMA *is less efficient than the tree-code* and the theoretical expectation that its CPU-time were linear in N is not confirmed by our results.

To conclude we can say that the efforts for the parallelization have to be done mainly on *tree-codes*, that appear to be more efficient than the FMA in single-processor runs. This does not mean that FMA is not an efficient algorithm. Infact, being easy adding high order multipole term, it is able to reach accuracies higher than classical *tree-codes*, which have been conceived to work up to second order term. Anyway higher accuracy is out of the purposes

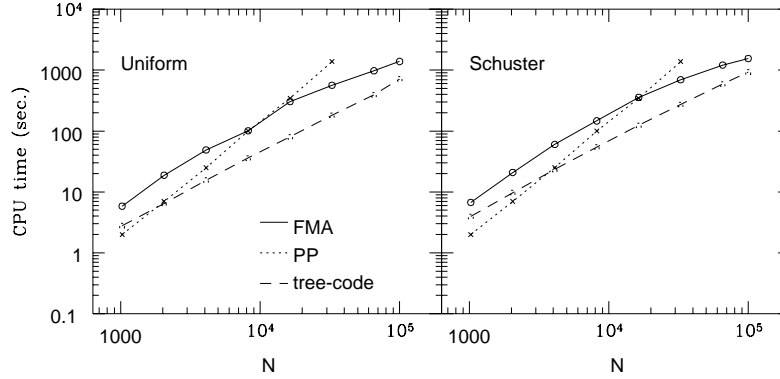


Figure 1: *CPU-time vs the number of particles in both cases*

of astrophysical simulations where, rather, it is important to deal adequately with a large range of *dynamical time scales*.

References

1. J. Barnes and P. Hut, *Nature*, *324*, (1986) 446.
2. R. Capuzzo-Dolcetta and R. Di Lisio, submitted to *J. of Comp. Phys.* (1996)
3. R. Capuzzo-Dolcetta and P. Miocchi, submitted to *J. of Comp. Phys.* (1996)
4. R. Capuzzo-Dolcetta and P. Miocchi, in *Science and Supercomputing at CINECA*, 1995 Report (Tecnoprint, Bologna, 1996) p.37
5. L. Greengard, in *The Rapid Evaluation of Potential Fields in Particle Systems*, PhD Thesis (MIT Press, Cambridge, MA, London, UK, 1987).
6. L. Hernquist, *Astroph. J. Suppl. Ser.*, *64* (1987) 715.
7. P. Miocchi, Graduation Thesis (University of L'Aquila, Italy, 1994).
8. J.J. Monaghan, *Ann. Rev. of Astron. & Astroph.*, *30* (1992) 542.
9. A. Schuster, *British Assoc. Report*, (1883) 427.